7231B

APPLICATION BRIEF

INTERFACING THE DYNAMIC RAM CONTROLLER TO THE iAPX 186

Jim Sleezer

1.0 INTRODUCTION

The 80186 microprocessor has integrated about 20 typically used system components into the same package as the microprocessor. This integration saves board space and design-in time. The 8208 Dynamic RAM Controller continues this system level integration. It is designed to control up to 256 Kbytes of Dynamic RAM (DRAM) using 64 K x 1 DRAMs, and up to 1 Mbyte using 256 K x 1 DRAMs.

Besides generating all DRAM control and timings, the 8208 allows various refresh types, frequencies, and microprocessor interfaces. Additionally, the 8208 does the 8 DRAM warm-up cycles back-to-back to prepare for operation.

By integrating the entire RAM timing and programmable refresh types, refresh rates, and interfaces into a single package, the user realizes significant savings in development time and board space. For example, a quick comparison of the 8208 versus a TTL implementation (using just the DRAM timing logic from Intel's iSBC012B memory board) yielded the following results:

1) a reduction in board space (10 in^2 to 3 in^2),

2) a reduction in power (1.2 A to 300 mA), and

3) much less design time (1 day).

The difference would be greater still if RAM warm-up, refresh, and interface programmability were added to the TTL implementation.

This Application Note will examine an 8208 to 80186 design. The reader should already have read the 8208 Data Sheet, the 80186 Data Sheet, and a DRAM Data Sheet*.

---

* While all DRAM references in this Application Note are based upon Intel's 2164A-15 64 K x 1 Dynamic RAM, any DRAM that meets the timing requirements in the Data Sheet, Table 8, and A.C. Characteristics, plus satisfies the Read Data Access Margin, will work.

2.0 HARDWARE DESIGN

An 8208 design can be divided into three areas: programming the 8208, DRAM compatibility, and system interface. While each topic will be covered in this Application Note, the 8208's programming logic defaults to an 8 MHz 80186 synchronous status interface with 150 ns access RAMs. All programming, RAM timings, and interface issues are satisfied for that configuration.

2.1.0 8208 PROGRAMMING

On the trailing edge of Reset, the 8208 samples the levels on two input pins and clocks in a 9 bit serial programming word. One input pin controls the type of refresh to be performed, while the other input pin alters the edge on which the 8208 samples memory commands. The program word further configures the 8208 for a refresh rate as a function of 8208 clock frequency, synchronous or asynchronous operation, and either an advanced acknowledge or Multibus compatible acknowledge.

2.1.1 REFRESH TYPES

If the REFRQ pin is sampled high at reset, an internal refresh timer is enabled; a low disables it. Both modes allow an external refresh cycle request by pulsing the REFRQ pin. An external request is generated by a low-to-high transition, and sampled by an 8208 (clock edge). Burst refresh occurs only when the timer is disabled and the REFRQ pin is sampled by two falling clock edges. The easiest method is to tie the REFRQ pin to Vcc (through a pull-up resistor); refresh cycles are transparent to the user.

2.1.2 8208 COMMANDS

The 8208 alters the point at which it samples a command and its response to the command inputs, based on the level sampled on PCTL when reset goes inactive. A high enables the status interface and a rising clock edge is used (this would be the middle of the T1 state; refer to the Timing Diagram). If low, the Multibus compatible interface is selected and a falling edge is used to allow for more propagation delay.

When the status interface is used, the status lines must be externally pulled up. The 80186 will tristate them when reset and the proper level (high) may not be seen by the 8208.

2.1.3 PROGRAM WORD

The program word defaults to a synchronous interface, fast acknowledge (for no wait states), and a refresh rate compatible with an 8 MHz clock (128 row/2 ms; 256 row/4 ms). When operating the 8208 at 8 MHz, most designs will not need to alter any programming bits and the PDI input pin can be tied to ground. If the 8208 is not run at 8 MHz a 74LS165-type shift register is needed to adjust for a proper refresh rate; otherwise, refresh cycles would not be performed often enough and data would be corrupted.

2.1.3.1 REFRESH RATE OPTIONS ($\overline{CIO}$, $\overline{CIT}$, $\overline{PLS}$, $\overline{FFS}$)

These four programming bits permit almost any DRAM to be used without wasting memory bandwidth. The combination of these four bits selects one of sixteEn clock intervals as shown in Table 1.

			Count Interval CI1, CI0 (8208 Clock Periods)			
CFS	PLS	FFS	00 (0%)	01 (10%)	10 (20%)	11 (30%)
0	1	1	118	106	94	82
0	0	1	59	53	47	41
0	1	0	74	66	58	50
0	0	0	37	33	29	25

Table 1. Refresh Count Intervals

The 8208 does not alter any other of its functions with these four bits. To determine which combination of bits to use, examine the following equation:

Equation 1.
Refresh Rate = count interval x 8208 clock period
14.6 usec = count interval x 190 ns
14.6 usec/.190 = 76.8 count interval

The next fastest Count Interval of 74 is chosen from Table 1. The bit configuration is: PLS = 1; FFS = 0; CI1 = 0; CI0 = 0, and generates seventy-four 8208 clocks between refresh cycles. A refresh cycle can be delayed up to one 8208 RAM cycle from the time it was requested to the time it is serviced. Thus, the 14.6 usec refresh rate is chosen to allow for these delays. The 190 ns clock period was picked at random. The refresh timer is restarted when the cycle is requested and not when the cycle begins executing. Note the difference in the sense of the programming bits. $\overline{PLS}$ = 0 is the same as PLS = 1. This notation is used throughout the Data Sheet.

2.1.3.3 INTERFACE OPTIONS ($\overline{S}$, $\overline{X}$)

The S programming bit adds synchronizers to the 8208's inputs when input signals cannot meet setup and hold times. The RD, WR inputs are still decoded as determined by PCTL, but these inputs will be sampled on a falling edge (status or command interface). The X bit allows either an 80186 (8086) no wait state acknowledge or an XACK (Multibus) type acknowledge. A synchronous interface should use the advanced acknowledge and an asynchronous interface the XACK acknowledge. XACK is removed by the inactive edge of RD or WR. If RD or WR goes inactive before the 8208 issues XACK, then no XACK is issued.

2.1.3.3 OTHER OPTIONS ($\overline{CFS}$, $\overline{RB}$, $\overline{RFS}$)

The CFS bit must be set to zero. This bit is reserved for future speed enhancements of the 8208. RFS has no effect on 8208 timings and may be set to either state. It is to be used with faster 8208's. RB is to allow for 32 bit wide memory arrays. If an 8 or 16 bit wide system is used, set this bit to its active state (RB = 0). The Bank Select pin must not select a RAM bank that is not physically present.

2.2 MICROPROCESSOR INTERFACE

The 8208's timings are optimized for an 8086 and 80186 system. The synchronous status interface offers the best performance (i.e., no wait states) and is the easiest to implement.

2.3 DRAM COMPATIBILITY

Table 2 lists the equations to determine whether a particular DRAM will work with the 8208. Four other questions are listed in the A.C. Characteristics Section in the 8208 Data Sheet.

Parameter	Rd. RF Cycles	Notes
tRP	2TCLCL—T26	1
tCPN	2.5TCLCL—T35	1
tRSH	3TCLCL—T34	1
tCSH	3TCLCL—T26	1
tCAH	2TCLCL—T34	1
tAR	2TCLCL—T26	1
tT	3/30	2
tRC	4TCLCL	1
tRAS	2TCLCL—T26	1
tCAS	3TCLCL—T34	1
tRCS	1.5TCLCL—TCL—T36—TBUF	1
tRCH	0.5TCLCL—T34	1

Parameter	WR Cycles	Notes
tRP	2TCLCL—T26	1
tCPN	2.5TCLCL—T35	1
tRSH	3TCLCL—T34	1
tCSH	4TCLCL—T26	1
tCAH	2TCLCL—T34	1
tAR	3TCLCL—T26	1
tT	3/30	2
tRC	6TCLCL	1
tRAS	4TCLCL—T26	1
tCAS	TCLCL—T34	1
tWCH	3TCLCL—T34	1, 3
tWCR	4TCLCL—T26	1, 3
tWP	4TCLCL—T36—TBUF	1
tRWL	4TCLCL—T36—TBUF	1
tCWL	4TCLCL—T36—TBUF	1
TWCS	TCLCL—T36—TBUF	

Table 2. DRAM Equations

These equations merely determine if the 8208 will provide proper margins for a DRAM. Whether a RAM works properly in a system is another issue. The Hardware Design Example section examines most of the important system timings.

3.0 HARDWARE DESIGN EXAMPLE

The objective is to have the 80186 run without wait states when accessing a DRAM array. The total amount of DRAM is 128K bytes and will be organized as 1 bank of 64K words.

Figure 1 is a block diagram of our design showing all relevant logic. The programming shift register is not needed since the 8208 will be operating at 8 MHz, and the other defult values are required. A data buffer is required in a no wait state design, since during reads the 8208 CAS line drives data onto the bus up to 50 ns past the end of T4. If another bus cycle were starting, then the multiplexed address/data lines would conflict with the driven data bus. This would reduce the systems' address to ALE setup margins. Figure 2 is a timing diagram of the design.

The timing parameters that are examined ensure that this portion of the system will operate properly. The parameters are:

1. Command setup and hold margin.
2. Address setup and hold margin.
3. Acknowledge setup and hold margin.
4. Write data setup and hold margin.
5. Read access margin.

3.1 ACKNOWLEDGE SETUP AND HOLD MARGIN

The 8208 early acknowledge (AACK) is intended to be connected to the SRDY input on the 80186 after being inverted. The AACK is issued at the beginning of T2 and must be valid at the beginning of T3.

1TCLCL - 8208 TCLAKL max - 7410 tPLH @ 15 pf - 80186 TSRYCL min _ 0

125 ns - 35 - 22 - 35
= 33 ns

The 80186 hold requirements, TCLSRY, of 15 ns is always met. The 15 ns hold time applies only when READY is being looked at by the 80186. Transitions that occur anywhere else in the bus cycle have no effect. AACK is two clocks long and is issued from a falling clock edge. AACK would always be sampled one clock into its duration. There would be a hold time of about 1 clock.

3.2 COMMAND SETUP AND HOLD MARGIN

Two events must occur for the 8208 to recognize a valid memory command. The 80186 status outputs are sampled by a rising clock edge (middle of T1 typically) and PE is sampled on the very next falling clock edge. If PE is not sampled at this point, no memory cycle will start. The status lines would have to go inactive before requesting another memory cycle.

The status setup margin is referenced to the middle of T4 or TI, and is required to be valid by the middle of T1.

1TCHCH - 80186 TCHSV max - 8208 TKVCH min _ 0

125 ns - 55 ns - 20 ns
= 50 ns

PE setup margin is referenced to the beginning of T1 and must be valid by the end of T1. PE selects the 8208 for a valid address range. It can be generated from either the address bus or using the 80186's programmable chip selects.

1 TCLCL - 80186 TCLCSV max - 8208 TPEVCL min _ 0

125 - 66 - 30
= 29 ns

Both PE and the RD, WR, and PCTL inputs require a 0 ns hold time to their respective clock edges.

The 8208 latches this information internally for cases when a refresh cycle delays a memory cycle from starting. Thus, a cycle will start when the refresh cycle finishes, even if the status signals have gone inactive. The hold margin is always met.

3.3 ADDRESS SETUP AND HOLD MARGINS

The 8208 requires the addresses to be stable before RAS goes active, and to remain stable for two clock periods thereafter. Unused address inputs should be pulled up to Vcc with a resistor.

The 8208 generates a margin of 0 ns minimum for the DRAM specification tASR when the 8208 specification TAVCL is met. If some DRAM is found that needs a more positive margin for tASR, then this requirement must be added to TAVCL.

The setup margin is between the clock edge that addresses are issued from to the 8208 issuing RAS, minus delays.

1 TCLCL + 8208 TCLRSL min[1] (@ 150 pf) - 80186 TCLAV max - 8282 IVOV max (@ 300 pf) - [8208 TAVCL min + DRAM tASR] $\geq$ 0

125 ns + 0 - 44 - 30 - (35 + 0)
= 16 ns

The 8208's address bus is divided into two halves. AL0-8 becomes the DRAM row address outputs and AH0-8 becomes the column addresses (64K DRAMs would need AL0-7 and AH0-7 connected to the address bus, AL8, AH8 would be tied to Vcc). Internally, the 8208 latches AH0-8 with CAS to provide for tCAH - column address hold time. This latching occurs near the end of T2 for read cycles and near the end of T3 for write cycles. When the RAM cycle is delayed due to refresh, the timing of AACK will ensure the two clock hold requirement. No equation is provided since this happens internally.

---

[1] Since this is not specified, 0 will be used for analysis only. Based upon design information this value would be about 20 ns.

3.4 WRITE DATA SETUP AND HOLD MARGIN

During write cycles, data from the 80186 must be valid at the DRAM when CAS goes low, and satify the DRAM tDS specification. Data must then be held valid and referenced to CAS long enough to meet the DRAM specification tDH. In this design example DEN is the limiting factor in the data setup margin. DEN is active before data is issued by the microprocessor, but there is a significant delay before the buffer is active. The result is that write data will be valid at the buffer before it is fully capable of transmitting data. The margin is referenced to the clock edge that issues DEN and the clock edge that issues CAS, minus delays.

TCHCL + 1 TCLCL = 8208 TCLCSL min (@ 150 pf) -
80186 TCVCTV max - 74LS245 TPZH max - DRAM TDS _ 0

55 + 125 + 62.5 - 70 - 40 - 0
= 132 ns

The hold margin is referenced to the edge that issues CAS and when valid data disappears. DEN is the controlling signal because it can go inactive before the data bus is floated by the microprocessor.

1 TCLCL + 1 TCLCH + 80186 TCVCTX min + 74LS245 TPLZ min[1] -
8208 TCLCSL max (@ 150 pf) - DRAM TDH _ 0

125 ns + 55 + 10 + 7.5 - 121 - 30
= 46.5 ns

The WE pulse length may cause problems with back-to-back bus cycles. Shortening the pulse width will not cause any other problems. The easiest solution is to factor in a shorter width signal, such as AACK, as is done in the design example.

---

[1] This parameter is not specified. For analysis, either assume 0 ns or use a more realistic value, such as one-half of typical.

3.5 READ DATA ACCESS MARGIN

The design example requires a buffer in the data path because the 8208 will not stop driving data onto the bus until after the end of T4. With back-to-back bus cycles this would cause bus contention and reduce address to ALE setup margins. The DRAM access parameter used is called "TCAC", and is referenced from the CAS active edge - not RAS. This parameter varies widely between manufacturers. When analyzing read access margins, some trade-off between buffer speed and TCAC delays must be considered.

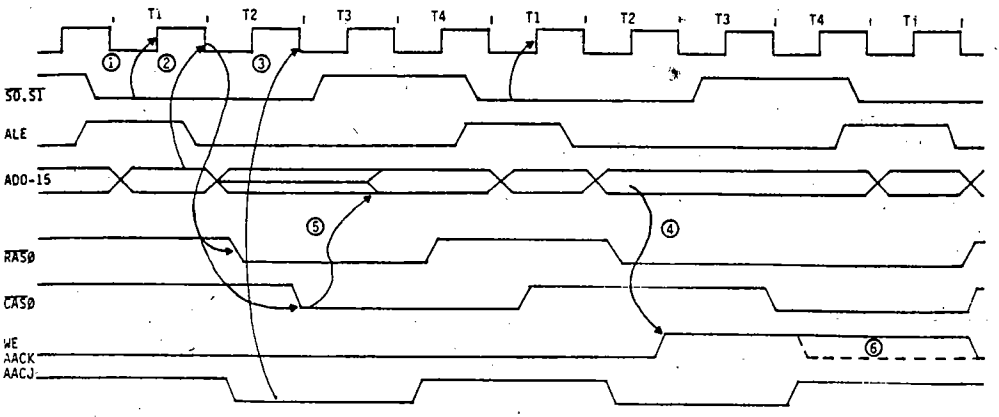
The 8208 starts a memory cycle, typically, at the end of T1, and data must be valid at the end of T3. With [refresh cycle] delayed bus cycles, data would still have to be valid in two clocks. The timing of the AACK signal guarantees this. From this two clock margin, buffer delays, TCAC delays, and others must be subtracted.

2 TCLCL - 8208 TCLCSL max (@ 150 pf) - DRAM TCAC max (@100 pf) - buffer delays max - 80186 TDVCL min _ 0

250 ns - 121 - 85 - 12 - 20
= 12 ns

4.0 SUMMARY

The 8208 solves most of the many design issues faced when adding a dynamic RAM array by giving the designer options. Options for various types of DRAMs, clock speeds, and system configurations. The margins that were just examined showed that the 8208 has plenty of margin in a system. Several margins were even higher. The READ DATA ACCESS MARGIN, for example, is considerably greater. The access time for DRAMS is specified with 100 pf loads, yet this was not added into the equation. Each designer should verify this analysis as specifications from manufacturer's change, without notice.



8208 Timing Diagram

Read then Write Cycle with one Bank of RAM

1. Status setup margin
2. Address setup margin
3. Acknowledge setup margin
4. Write data setup margin
5. Read access margin: based upon TCAC
6. Shortened WE pulse